

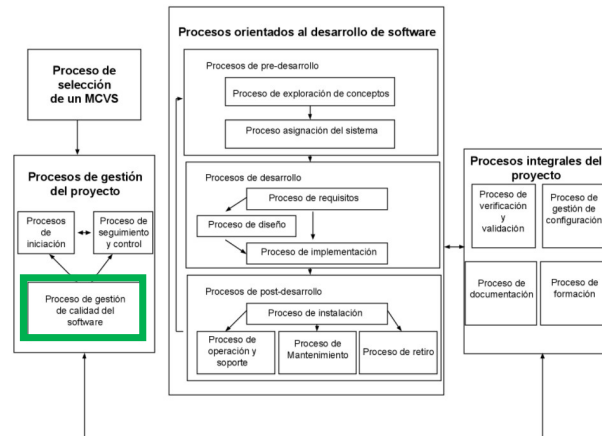


# Calidad

Introducción a la administración de proyectos



## Introducción – Repaso IEEE 1074



Introducción a la administración de proyectos



# Calidad

Todo lo que hacemos en éste ámbito es en pos de  
buscar construir aplicaciones de calidad

Introducción a la administración de proyectos

1. ¿Para qué vienen a la universidad?
2. Todo lo que hacemos en este ámbito es en pos de buscar construir aplicaciones de calidad, pero ¿qué es la calidad de software?



Lo que está claro a partir de estas definiciones es que la calidad es algo **relativo**. **Siempre va a depender de los requisitos o necesidades** que se desee satisfacer. Por eso la evaluación de la calidad de un producto siempre **va a implicar una comparación requisitos preestablecidos y el producto desarrollado**.

El problema es que, por lo general, una parte de los requisitos van a estar explícitos (se encontrarán en la especificación de requisitos de software, tanto los funcionales como otros requisitos), pero otra parte van a quedar implícitos (el usuario sabe lo que quiere pero no siempre es capaz de expresarlo). Hay que intentar que queden implícitos la menor cantidad de requisitos posible. No se podrá conseguir un producto de buena calidad sin una buena especificación de requisitos de software.

No es suficiente hablar por hablar diciendo que la calidad de software es importante, **es importante definir explícitamente lo que significa calidad de software**.

En el desarrollo del software, la calidad del diseño incluye el grado en el que el diseño cumple las funciones y características especificadas en el modelo de requerimientos. La calidad de la conformidad se centra en el grado en el que la implementación se apega al diseño y en el que el sistema resultante cumple sus metas de requerimientos y desempeño.

Pero, ¿son la calidad del diseño y de la conformidad los únicos aspectos que deben considerar los ingenieros de software? Robert Glass afirma que es mejor plantear una relación más intuitiva:

satisfacción del usuario = producto que funciona + buena calidad + entrega dentro del presupuesto y plazo

En última instancia, Glass sostiene que **la calidad es importante, pero que si el usuario no está satisfecho, nada de lo demás importa.**



## Calidad de Software

**“Proceso eficaz de software** que se aplica de manera que crea un **producto útil** que proporciona **valor** medible a quienes lo producen y a quienes lo utilizan.”

Introducción a la administración de proyectos

Hablando específicamente de Calidad de software, Elegimos la siguiente definición:  
**“Proceso eficaz de software** que se aplica de manera que crea un **producto útil** que proporciona **valor** medible a quienes lo producen y a quienes lo utilizan.”

1. **Un proceso eficaz de software establece la infraestructura que da apoyo a cualquier esfuerzo** de elaboración de un producto de software de alta calidad. Los aspectos de administración del proceso generan las verificaciones y equilibrios que ayudan a evitar que el proyecto caiga en el caos, contribuyente clave de la mala calidad. Las prácticas de ingeniería de software permiten al desarrollador analizar el problema y diseñar una solución sólida, ambas actividades críticas de la construcción de software de alta calidad. Por último, las actividades sombrilla (integrales), tales como administración del cambio y revisiones técnicas, tienen tanto que ver con la calidad como cualquier otra parte de la práctica de la ingeniería de software.
2. **Un producto útil entrega contenido, funciones y características que el usuario final desea;** sin embargo, de igual importancia es que entrega estos activos en forma confiable y libre de errores. Un producto útil siempre satisface los requerimientos establecidos en forma explícita por los participantes. Además,

satisface el conjunto de requerimientos (por ejemplo, la facilidad de uso) con los que se espera que cuente el software de alta calidad.

3. ***Al agregar valor para el productor y para el usuario de un producto, el software de alta*** calidad proporciona beneficios a la organización que lo produce y a la comunidad de usuarios finales. La organización que elabora el software obtiene valor agregado porque el software de alta calidad requiere un menor esfuerzo de mantenimiento, menos errores que corregir y poca asistencia al cliente. Esto permite que los ingenieros de software dediquen más tiempo a crear nuevas aplicaciones y menos a repetir trabajos mal hechos. La comunidad de usuarios obtiene valor agregado porque la aplicación provee una capacidad útil en forma tal que agiliza algún proceso de negocios. El resultado final es 1) mayores utilidades por el producto de software, 2) más rentabilidad cuando una aplicación apoya un proceso de negocios y 3) mejor disponibilidad de información, que es crucial para el negocio.



## Calidad de software

- Calidad de producto
- Calidad de proceso

Introducción a la administración de proyectos

A la hora de definir la calidad del software se pueden adoptar diferentes enfoques.

Como primer paso es importante diferenciar entre la calidad del producto software y la calidad del proceso de desarrollo. No obstante, las metas que se establezcan para la calidad del producto van a determinar las metas a establecer para la calidad del proceso de desarrollo, ya que la calidad del producto va a estar en función de la calidad del proceso de desarrollo. Sin un buen proceso de desarrollo es casi imposible obtener un buen producto.





## Calidad de producto

El desafío mas grande a la hora de hablar de la calidad del producto es encontrar un conjunto de propiedades que nos den una indicación de su calidad

Introducción a la administración de proyectos

La calidad del producto software se diferencia de la calidad de otros productos de fabricación industrial, ya que el software tiene ciertas características especiales:

- **El software es un producto mental;** no restringido por las leyes de la Física o por los límites de los procesos de fabricación. Es algo abstracto, y su calidad también lo es.
- **Se desarrolla, no se fabrica.** El costo está fundamentalmente en el proceso de diseño, no en la producción. Y los errores se introducen también en el diseño, no en la producción.
- **El software no se deteriora con el tiempo.** No es susceptible a los defectos del entorno, y su curva de fallos es muy diferente de la del hardware. Todos los problemas que surjan durante el mantenimiento estaban allí desde el principio, y afectan a todas las copias del mismo; no se generan nuevos errores.
- **Es artesanal en gran medida.** El software, en su mayoría, se construye a medida, en vez de ser construido ensamblando componentes existentes y ya probados, lo que dificulta aún más el control de su calidad. Aunque se ha escrito mucho sobre la reutilización del software, hasta ahora se han conseguido pocos éxitos tangibles.
- **El mantenimiento del software** es mucho más complejo que el mantenimiento del hardware. Cuando un componente hardware se deteriora se sustituye por

una pieza de repuesto, pero cada fallo en el software implica un error en el diseño o en el proceso mediante el cual se tradujo el diseño en código máquina ejecutable.

- **Es engañosamente fácil realizar cambios** sobre un producto software, **pero los efectos de estos cambios se pueden propagar** de forma explosiva e incontrolada.
- Como **disciplina**, el desarrollo de software es aún muy **joven**, por lo que las técnicas de las que disponemos aún no son totalmente efectivas o no están totalmente calibradas.



## Calidad del proceso

La mejora de la calidad del software: ¿cómo utilizar la información disponible acerca de la calidad del producto software para mejorar su calidad a lo largo del ciclo de vida? No sólo es posible «medir» la calidad, sino también «construir» la calidad durante el proceso de desarrollo del producto

Introducción a la administración de proyectos

**Gestión de calidad:** determinación y aplicación de las políticas de calidad de la empresa (objetivos y directrices generales)

**Garantía o aseguramiento de calidad:** conjunto de actividades planificadas y sistemáticas necesarias para proporcionar confianza en que el producto software satisfará los requisitos dados de calidad



## Modelos de calidad



Introducción a la administración de proyectos

Los modelos de calidad del software vienen a ayudar en la puesta en práctica del concepto general de calidad que vimos en el apartado anterior, ofreciendo una definición operacional.

En los modelos de calidad, la calidad se define de forma jerárquica. Es un concepto que se deriva de un conjunto de subconceptos, cada uno los cuales se va a evaluar a través de un conjunto de indicadores o métricas.

Tienen una estructura, por lo general, en tres niveles:

- En el nivel más alto de la jerarquía se encuentran los **factores de calidad**, que representan la calidad desde el punto de vista del usuario. Son las características que componen la calidad. También se les llama **atributos de calidad externos**.
- Cada uno de los factores se descompone en un conjunto de **criterios de calidad**. Son atributos que, cuando están presentes, contribuyen al aspecto de la calidad, que el factor asociado representa. Se trata de una visión de la calidad desde el punto de vista del producto software. También se les llama **atributos de calidad internos**.
- Para cada uno de los criterios de calidad se definen entonces un conjunto de **métricas**, que son medidas cuantitativas de ciertas características del producto que, cuando están presentes, dan una indicación del grado en que dicho producto posee un determinado atributo de calidad.

La ventaja de los modelos de calidad es que la calidad se convierte en algo concreto, que se puede definir, que se puede medir y, sobre todo, que se puede planificar.



## Factores de la calidad



McCall propone una clasificación útil de los factores que afectan la calidad del software. Éstos se ilustran en esta figura y se centran en tres aspectos importantes del producto de software: sus **características operativas**, su **capacidad de ser modificado** y su **adaptabilidad a nuevos ambientes**.

**Corrección.** Grado en el que un programa satisface sus especificaciones y en el que cumple con los objetivos de la misión del cliente.

**Confiabilidad.** Grado en el que se espera que un programa cumpla con su función y con la precisión requerida.

**Usabilidad.** Esfuerzo que se requiere para aprender, operar, preparar las entradas e interpretar las salidas de un programa.

**Integridad.** Grado en el que es posible controlar el acceso de personas no autorizadas al software o a los datos.

**Eficiencia.** Cantidad de recursos de cómputo y de código requeridos por un programa para llevar a cabo su función.

**Facilidad de recibir mantenimiento.** Esfuerzo requerido para detectar y corregir un error en un programa

**Flexibilidad.** Esfuerzo necesario para modificar un programa que ya opera.

**Susceptibilidad de someterse a pruebas.** Esfuerzo que se requiere para probar un programa a fin de garantizar que realiza la función que se pretende.

**Portabilidad.** Esfuerzo que se necesita para transferir el programa de un ambiente de sistema de hardware o software a otro.

**Reusabilidad.** Grado en el que un programa (o partes de uno) pueden volverse a utilizar en otras aplicaciones.

**Interoperabilidad.** Esfuerzo requerido para acoplar un sistema con otro.

Tanto los factores de la calidad de software como las dimensiones de la calidad de software, son enfoques **cualitativos** de la calidad, lo cual hace del análisis de esta una actividad subjetiva.

La comunidad de software ha desarrollado mecanismos para analizar cuantitativamente las diferentes manifestaciones de la calidad (no es posible medir la calidad por sí misma, si no hacemos mediciones indirectas). Esto lo veremos en la unidad de métricas.



## Criterios de calidad

Usabilidad	• Facilidad de operación • Facilidad de aprendizaje
Integridad	• Control de accesos • Facilidad de auditoría
Corrección	• Completitud • Consistencia
Confiabilidad	• Consistencia • Tolerancia a fallos
Eficiencia	• Eficiencia en ejecución • Eficiencia en almacenamiento
Facilidad de mantenimiento	• Modularidad • Consistencia
Facilidad de prueba	• Modularidad • Simplicidad
Flexibilidad	• Capacidad de expansión • Generalidad
Reusabilidad	• Independencia del hardware • Independencia Sistema / Software
Interoperabilidad	• Compatibilidad de comunicaciones • Compatibilidad de datos
Portabilidad	• Independencia Sistema / software • Independencia del hardware

Introducción a la administración de proyectos

Algunos criterios (atributos internos del producto) asociados a cada factor de calidad:

### Usabilidad

- Facilidad de operación
- Facilidad de aprendizaje

### Integridad

- Control de accesos
- Facilidad de auditoría

### Corrección

- Completitud
- Consistencia

### Confiabilidad

- Consistencia
- Tolerancia a fallos

### Eficiencia

- Eficiencia en ejecución
- Eficiencia en almacenamiento

### Facilidad de mantenimiento

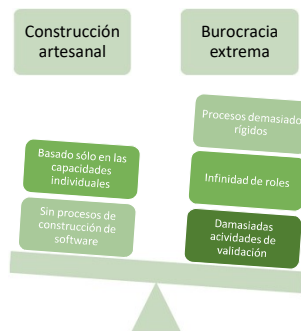
- Modularidad



- Consistencia
- Facilidad de prueba
- Modularidad
- Simplicidad
- Flexibilidad
  - Capacidad de expansión
  - Generalidad
- Reusabilidad
  - Independencia del hardware
  - Independencia Sistema / Software
- Interoperabilidad
  - Compatibilidad de comunicaciones
  - Compatibilidad de datos
- Portabilidad
  - Independencia Sistema / software
  - Independencia del hardware



## El dilema de la calidad



Introducción a la administración de proyectos

En un extremo tenemos la construcción artesanal, sin procesos de software definidos, y del otro lado tenemos procesos de construcción de software complejos, con muchísimas tareas a realizar. El punto medio y de equilibrio es el desarrollo de un software “suficientemente bueno”.

Exactamente, ¿qué significa “suficientemente bueno”? El software suficientemente bueno contiene las funciones y características de alta calidad que desean los usuarios, pero al mismo tiempo tiene otras más oscuras y especializadas que contienen errores conocidos. El vendedor de software espera que la gran mayoría de usuarios finales perdone los errores gracias a que estén muy contentos con la funcionalidad de la aplicación.

Aparece el problema costo (ojo que la mala calidad también lo tiene).



## ¿Cómo se logra la calidad?

1. Métodos de la ingeniería del software
2. Administrar bien el proyecto
3. Realizar un control de calidad exhaustivo
4. Contar con infraestructura de aseguramiento de calidad

Introducción a la administración de proyectos

### **Métodos de la ingeniería de software**

Si espera construir software de alta calidad, debe entender el problema que se quiere resolver. También debe ser capaz de crear un diseño que esté de acuerdo con el problema y que al mismo tiempo tenga características que lleven al software a las dimensiones y factores de calidad ya vistos. Es necesario conocer y aplicar los diferentes métodos para el análisis y diseño exitoso para la construcción de software.

### **Técnicas de administración de proyectos**

El efecto de las malas decisiones de administración sobre la calidad del software ya lo hemos mencionado. Las implicaciones son claras:

- 1) Si un gerente de proyecto usa estimaciones para verificar que las fechas pueden cumplirse,
- 2) se comprenden las dependencias de las actividades programadas y el equipo resiste la tentación de usar atajos,
- 3) la planeación del riesgo se lleva a cabo de manera que los problemas no alienten el caos, entonces la calidad del software se verá influida de manera positiva.

Además, el plan del proyecto debe incluir técnicas explícitas para la administración de la calidad y el cambio.

### **Control de calidad**

El control de calidad incluye un conjunto de acciones de ingeniería de software que ayudan a asegurar que todo producto del trabajo cumpla sus metas de calidad. Los modelos se revisan para garantizar que están completos y que son consistentes. El código se inspecciona con objeto de descubrir y corregir errores antes de que comiencen las pruebas. Se aplica una serie de etapas de prueba para detectar los errores en procesamiento lógico, manipulación de datos y comunicación con la interfaz. La combinación de mediciones con retroalimentación permite que el equipo del software sintonice el proceso cuando cualquiera de estos productos del trabajo falla en el cumplimiento de las metas de calidad. Las actividades de control de calidad las veremos mas adelante.

### **Aseguramiento de la calidad**

El aseguramiento de la calidad establece la infraestructura de apoyo a los métodos sólidos de la ingeniería de software, la administración racional de proyectos y las acciones de control de calidad, todo de importancia crucial si se trata de elaborar software de alta calidad. Además, el aseguramiento de la calidad consiste en un conjunto de funciones de auditoría y reportes para evaluar la eficacia y completitud de las acciones de control de calidad. La meta del aseguramiento de la calidad es proveer al equipo administrativo y técnico los datos necesarios para mantenerlo informado sobre la calidad del producto, con lo que obtiene perspectiva y confianza en que las acciones necesarias para lograr la calidad del producto funcionan. Por supuesto, si los datos provistos a través del aseguramiento de la calidad identifican los problemas, es responsabilidad de la administración enfrentarlos y aplicar los recursos necesarios para resolver los correspondientes a la calidad. Mas adelante hablaremos en mas detalle sobre el aseguramiento de la calidad.



## Control de la calidad

- El objetivo de las actividades de control de calidad es comprobar si un producto posee o no una determinada característica de calidad en el grado requerido.
- Actividades de control de calidad:
  - a. Controles Estáticos (Revisiones)
  - b. Controles Dinámicos (Prueba del software)

Introducción a la administración de proyectos

Técnicas de revisión o revisiones técnicas:

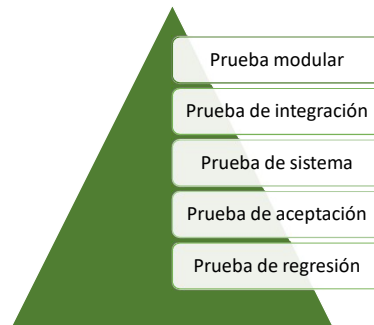
El objetivo principal de las revisiones técnicas es encontrar errores durante el proceso a fin de que no se conviertan en defectos después de liberar el software. El beneficio obvio de las revisiones técnicas es el descubrimiento temprano de los errores, de modo que no se propaguen a la siguiente etapa del proceso del software.

Varios estudios de la industria indican que las actividades de diseño introducen de 50 a 65 por ciento de todos los errores (y en realidad de todos los defectos) durante el proceso del software. Sin embargo, las técnicas de revisión han demostrado tener una eficacia de hasta 75 por ciento para descubrir fallas del diseño.

La diferencia principal entre los controles estáticos y dinámicos es que los segundos requieren la ejecución del objeto que se está controlando.



## Tipos de pruebas



Introducción a la administración de proyectos

Prueba modular  
Prueba de integración  
Prueba de sistema  
Prueba de aceptación  
Prueba de regresión

Métodos de prueba:  
Caja blanca  
Caja negra



## Aseguramiento de la calidad

Actividades específicas del proceso que permiten a una organización de software garantizar que hace “las cosas correctas en el momento correcto y de la forma correcta”

Introducción a la administración de proyectos

**SQA (Software Quality Assurance) / GSC (Garantía de Calidad de Software) engloba:**

- 1) Un enfoque de gestión de calidad;
- 2) Tecnología de ingeniería del software efectiva (métodos y herramientas);
- 3) Revisiones técnicas formales que se aplican durante el proceso del software;
- 4) Una estrategia de pruebas;
- 5) *El control de la documentación del software y de los cambios realizados;*
- 6) *Un procedimiento que asegure un ajuste a los estándares de desarrollo del software (cuando sea posible),*
- 7) Mecanismos de medición y de generación de informes.



## ¿Cuándo se hace?



Introducción a la administración de proyectos

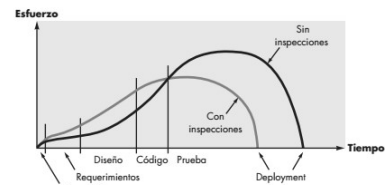
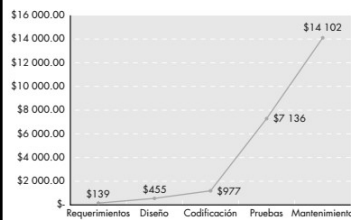
Algunos desarrolladores de software continúan creyendo que la calidad del software es algo en lo que empiezan a preocuparse una vez que se ha generado el código. ¡Nada más lejos de la realidad! La garantía de calidad del software es una actividad de protección que se aplica a lo largo de todo el proceso del software.

Las decisiones administrativas pueden afectar significativamente en la calidad del producto. Por ejemplo pensemos en los efectos de una mala estimación.





## ¿Cuándo se hace?



Introducción a la administración de proyectos

Entre más pronto se detecte un problema en el software, mejor. Y hay dos clases de problemas. Uno son los errores, que son problemas de implementación. El otro son las fallas del software: problemas de arquitectura en el diseño. La gente presta demasiada atención a los errores pero no la suficiente a las fallas.



## ¿Quién la hace?



Introducción a la administración de proyectos

¿Quién la hace? Todos los Stakeholders

El grupo de ACS debe ver el software desde el punto de vista del cliente. ¿Cumple el software con los factores de calidad?

El grupo de garantía de calidad es responsable de definir el tipo de control de calidad que se va a realizar, pero por lo general no es responsable de efectuar personalmente dichos controles, sino que estos serán realizados por los miembros del equipo de control de calidad para evaluar y mejorar el proceso de desarrollo para conseguir productos de más calidad.

Las principales tareas del grupo de garantía de calidad son:

1. **Planificación de la calidad:** consiste en seleccionar, clasificar y ponderar las propiedades de calidad que se van a establecer como requisitos, con respecto al producto y con respecto al proceso. Se elegirán también los mecanismos de control de calidad a utilizar para medir y evaluar esas características y se determinarán las metas a alcanzar.

2. **Supervisión de la calidad:** Consiste en supervisar y corregir, si es necesario, el trabajo que se está realizando (según los resultados obtenidos con las actividades de control de calidad), con el objetivo de llegar a satisfacer los requisitos establecidos.
3. **Construcción de la calidad:** Actividades constructivas son aquellas que sirven para –construir- la calidad, es decir, son actividades preventivas cuyo objetivo es evitar la introducción de errores mediante la puesta en práctica de ciertos principios, métodos, formalismos y herramientas.



## Aseguramiento de la calidad - Elementos

- Estándares
- Revisiones y auditorías
- Pruebas
- Colección y análisis de errores
- Administración del cambio
- Educación
- Administración de proveedores
- Administración de la seguridad
- Administración del riesgo

Introducción a la administración de proyectos

### **Estándares.**

El IEEE, ISO y otras organizaciones que establecen estándares han producido una amplia variedad de ellos para ingeniería de software y documentos relacionados. Los estándares los adopta de manera voluntaria una organización de software o los impone el cliente u otros participantes. El trabajo del ACS es asegurar que los estándares que se hayan adoptado se sigan, y que todos los productos del trabajo se apeguen a ellos.

### **Revisiones y auditorías.**

Las revisiones técnicas son una actividad del control de calidad que realizan ingenieros de software para otros ingenieros de software. Su objetivo es detectar errores. Las auditorías son un tipo de revisión efectuada por personal de ACS con objeto de garantizar que se sigan los lineamientos de calidad en el trabajo de la ingeniería de software. Por ejemplo, una auditoría del proceso de revisión se efectúa para asegurar que las revisiones se lleven a cabo de manera que tengan la máxima probabilidad de descubrir errores.

### **Pruebas.**

Las pruebas del software son una función del control de calidad que tiene un objetivo

principal: detectar errores. El trabajo del ACS es garantizar que las pruebas se planeen en forma apropiada y que se realicen con eficiencia, de modo que la probabilidad de que logren su objetivo principal sea máxima.

#### **Colección y análisis de los errores.**

La única manera de mejorar es medir cómo se está haciendo algo. El ACS reúne y analiza errores y datos acerca de los defectos para entender mejor cómo se cometen los errores y qué actividades de la ingeniería de software son más apropiadas para eliminarlos.

#### **Administración del cambio.**

El cambio es uno de los aspectos que más irrumpe en cualquier proyecto de software. Si no se administra en forma adecuada, lleva a la confusión y ésta casi siempre genera mala calidad. El ACS asegura que se hayan instituido prácticas adecuadas de administración del cambio.

#### **Educación.**

Toda organización de software quiere mejorar sus prácticas de ingeniería de software. Un contribuyente clave de la mejora es la educación de los ingenieros de software, de sus gerentes y de otros participantes. La organización de ACS lleva el liderazgo en la mejora del proceso de software y es clave para proponer y patrocinar programas educativos.

#### **Administración de los proveedores.**

Son tres las categorías de software que se adquieren a proveedores externos: paquetes contenidos en una caja (por ejemplo, Office, de Microsoft); un shell personalizado, que da una estructura básica, tipo esqueleto, que se adapta de manera única a las necesidades del comprador; y software contratado, que se diseña y construye especialmente a partir de especificaciones provistas por la organización cliente. El trabajo de la organización de ACS es garantizar que se obtenga software de alta calidad a partir de las sugerencias de prácticas específicas de calidad que el proveedor debe seguir (cuando sea posible) y de la incorporación de cláusulas de calidad como parte de cualquier contrato con un proveedor externo.

#### **Administración de la seguridad.**

Con el aumento de los delitos cibernéticos y de las nuevas regulaciones gubernamentales respecto de la privacidad, toda organización de software debe instituir políticas para proteger los datos en todos los niveles, establecer cortafuegos de protección para las webapps y asegurar que el software no va a ser vulnerado internamente. El ACS garantiza que para lograr la seguridad del software, se utilicen el proceso y la tecnología apropiados.

**Seguridad.**

Debido a que el software casi siempre es un componente crucial de los sistemas humanos (como aplicaciones automotrices o aeronáuticas), la consecuencia de defectos ocultos puede ser catastrófica. El ACS es responsable de evaluar el efecto de las fallas del software y de dar los pasos que se requieren para disminuir el riesgo.

**Administración de riesgos.**

Aunque el análisis y la mitigación de riesgos es asunto de los ingenieros de software, la organización del ACS garantiza que las actividades de administración de riesgos se efectúen en forma apropiada y que se establezcan planes de contingencia relacionados con los riesgos.



## Sistema de calidad

Es el que define cómo implementar la garantía o aseguramiento de la calidad

Introducción a la administración de proyectos

El tipo y número de actividades de garantía de calidad que es necesario adoptar en un proyecto o en una organización concreta depende mucho del tamaño y complejidad de los productos de software que se estén desarrollando. También influyen muchos otros factores, como pueden ser el tipo de proceso de desarrollo de software que se utiliza, la estructura organizativa de la empresa, la motivación del personal, los métodos y herramientas que se estén utilizando, etc.



## Resumen

Calidad: ¿Qué es?

Calidad de producto

Calidad de proceso

Modelos de calidad (factores y criterios)

Control de calidad (Revisiones estáticas y pruebas dinámicas)

Aseguramiento de la calidad: ¿Qué es? ¿Quién la hace? ¿Cuándo se hace? ¿Cómo se hace?

Introducción a la administración de proyectos





Introducción a la administración de proyectos